

# Lecture 09 - Deep Learning and CNNs

Prof. André Gustavo Hochuli

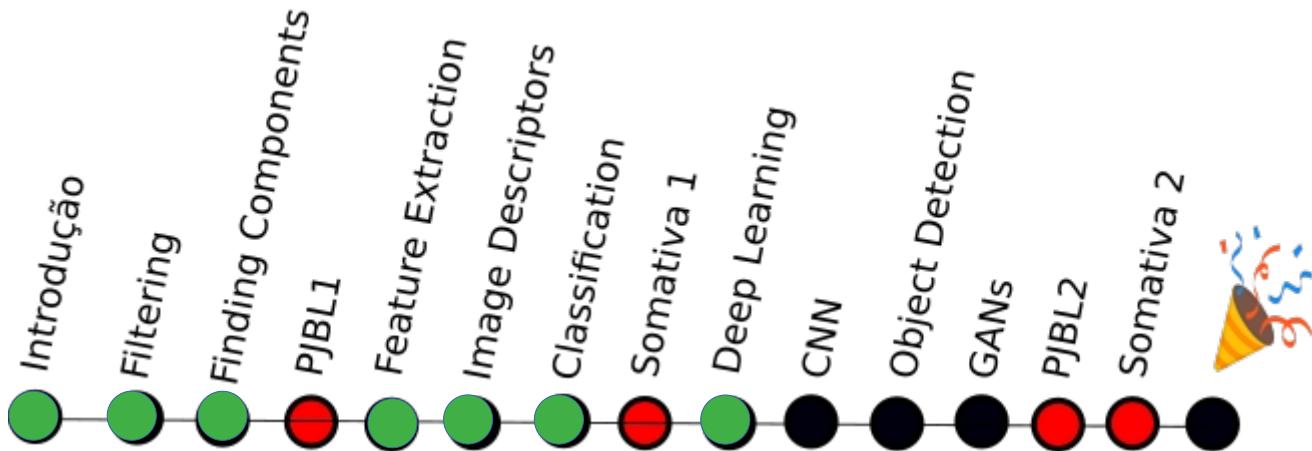
[gustavo.hochuli@pucpr.br](mailto:gustavo.hochuli@pucpr.br)

[aghochuli@ppgia.pucpr.br](mailto:aghochuli@ppgia.pucpr.br)

# Topics

## Deep Learning

- Convolutional Neural Network
  - Lenet
  - Imagenet
  - Deep Networks
- Practice



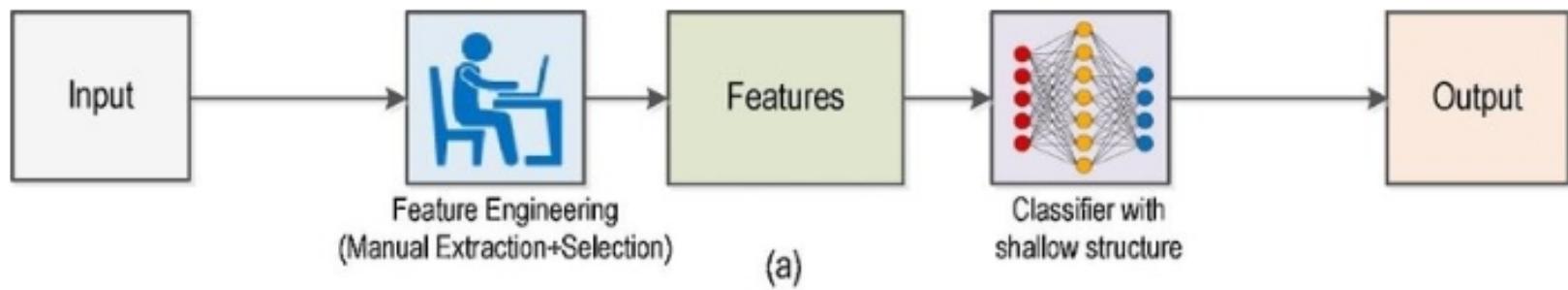
# Avaliação Institucional 2023.2



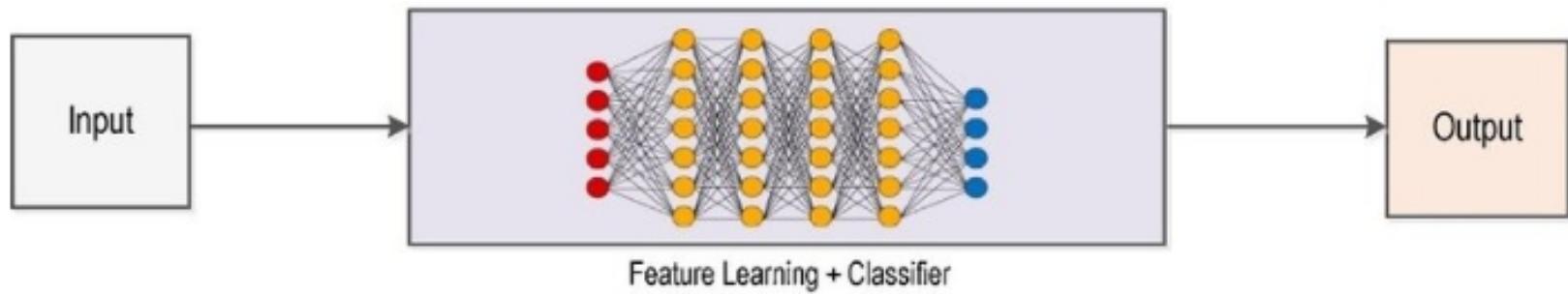
<https://pucpr.vouavaliar.com.br/login.zul>

# Traditional and Deep Learning

- Traditional

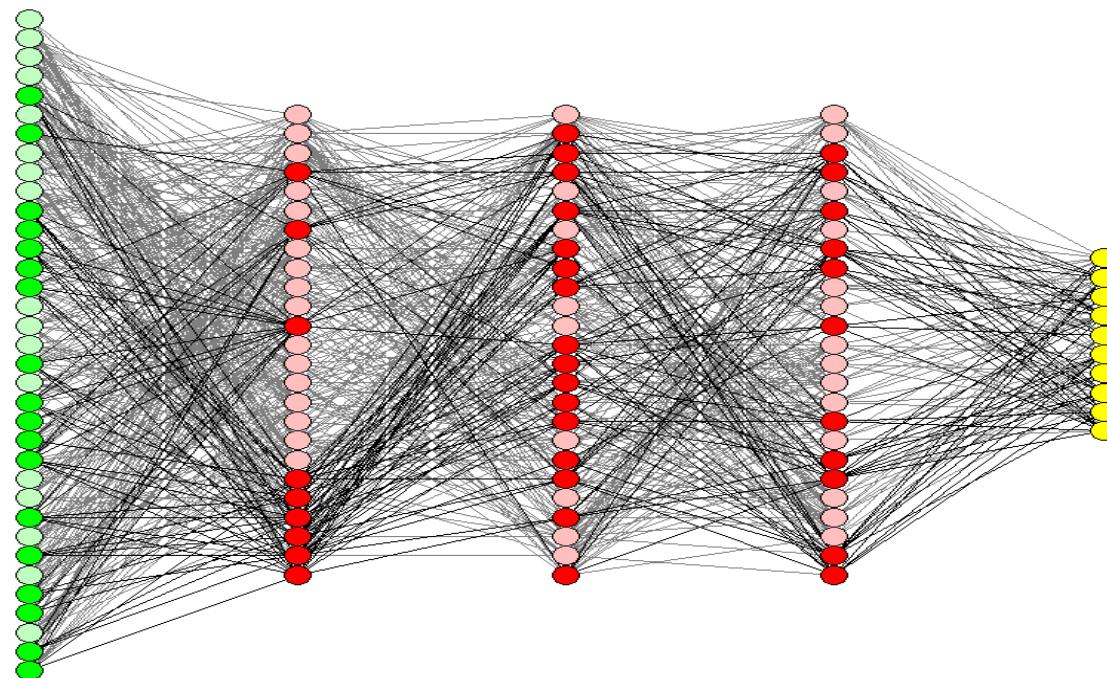


- Deep



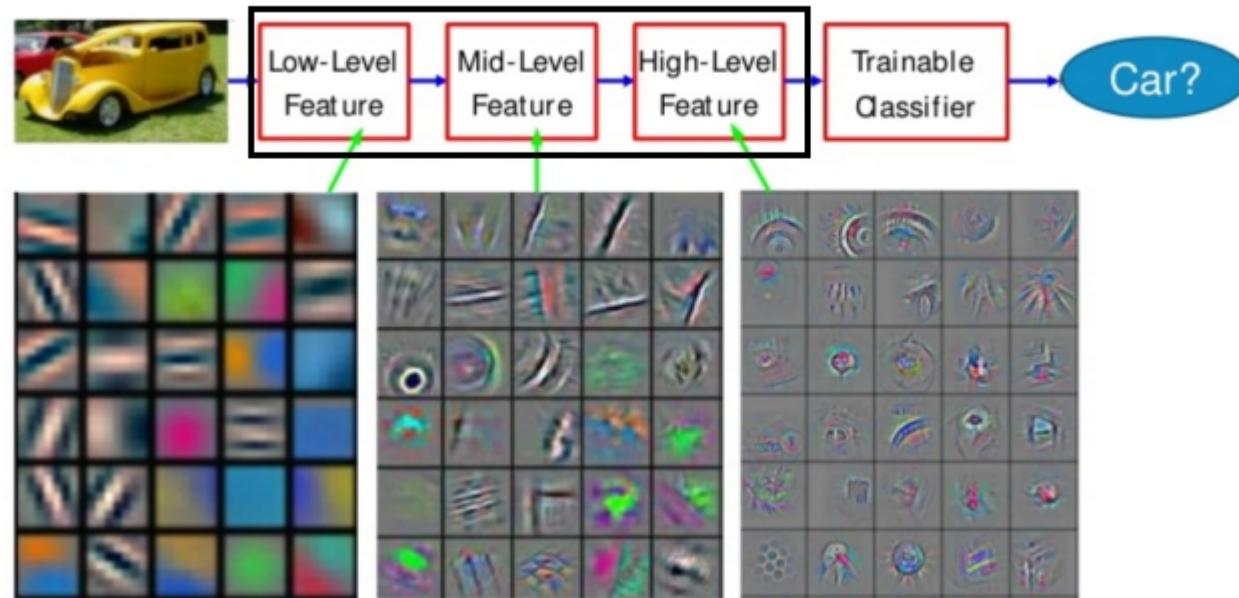
# Deep Learning

- Dense Multi-Layer Architectures



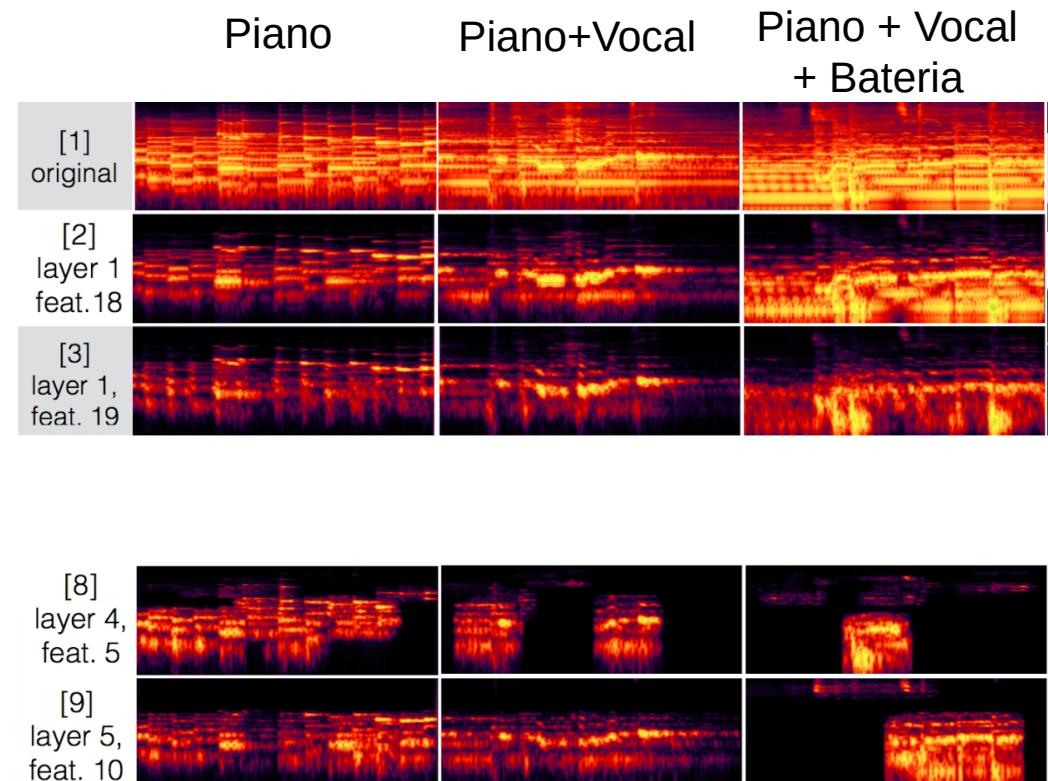
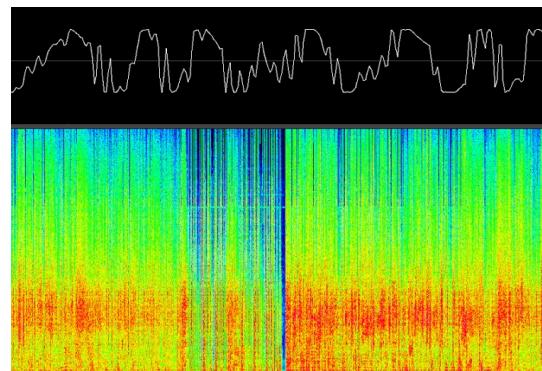
# Deep Learning

- Implicit feature extraction
- Learnable Filters
- Deep Abstraction



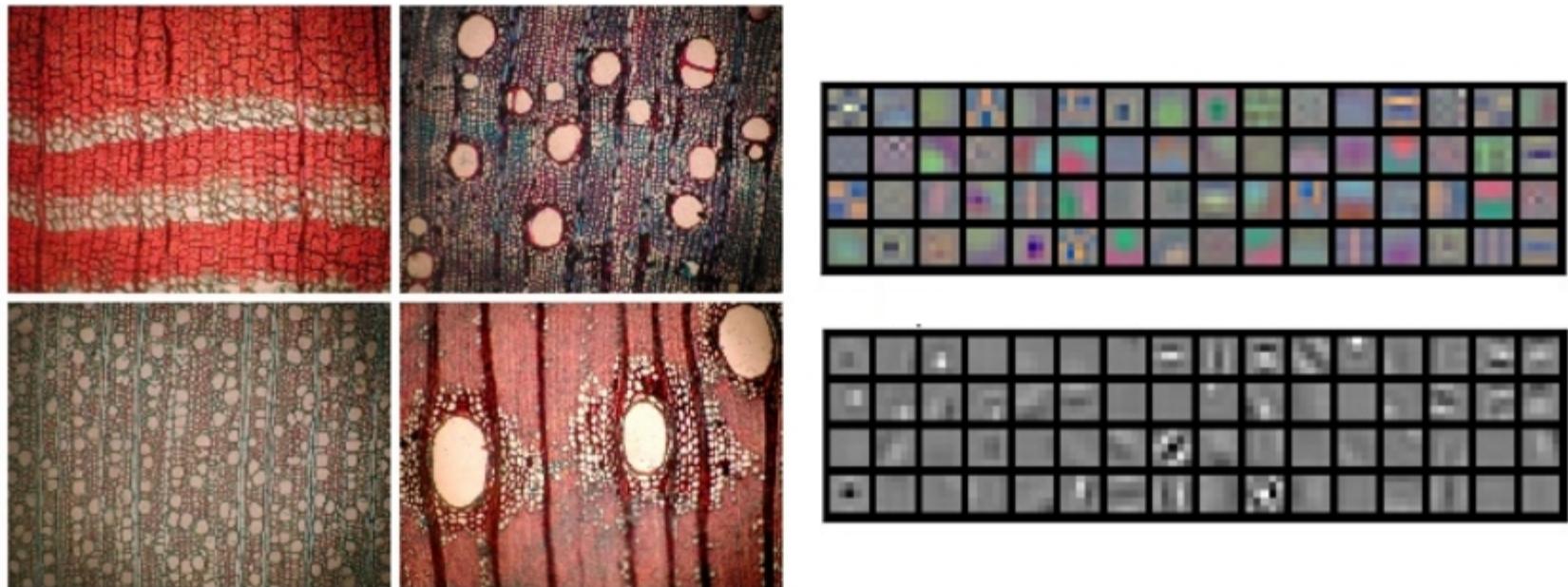
# Deep Learning

- Wide applied in the Computer Vision area (audio, images, video processing, etc.).



# Deep Learning

- Tissue Classification
- Medical Images



# Deep Learning

- Face

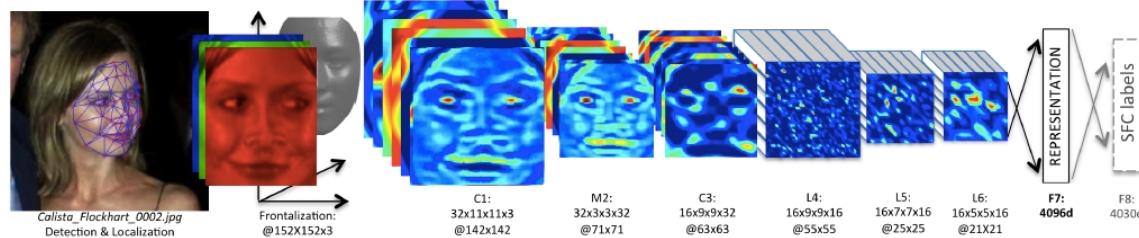


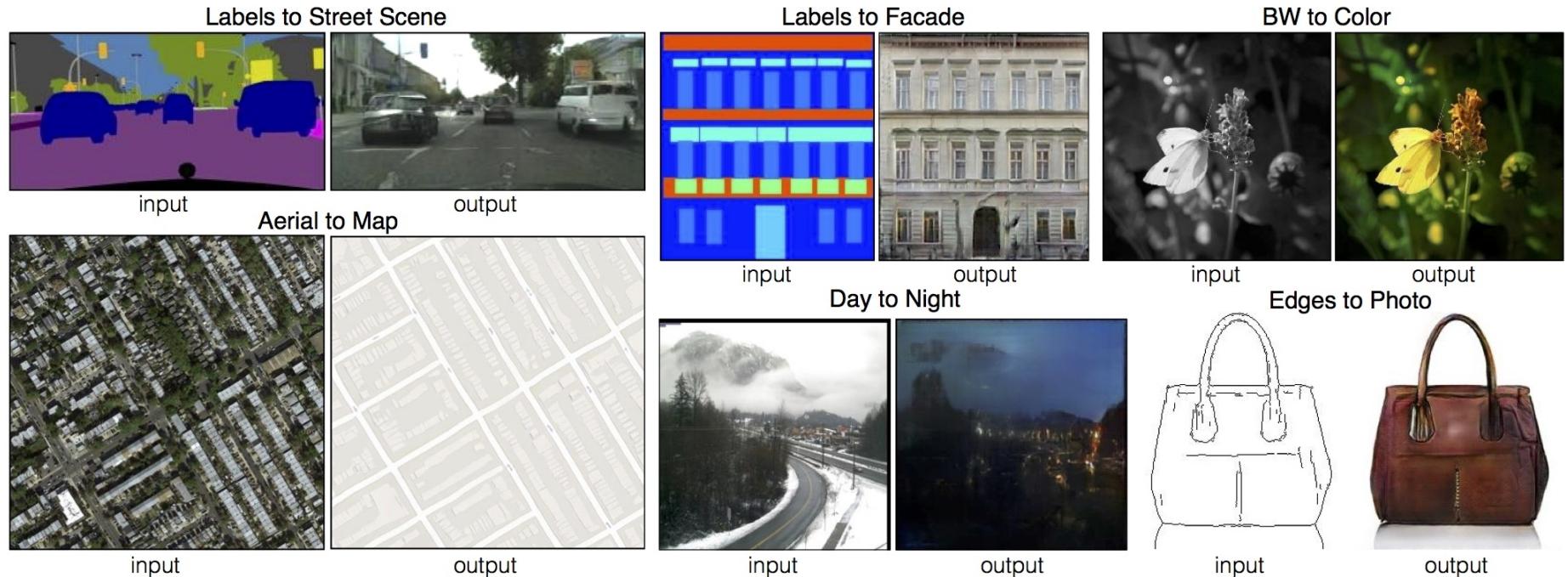
Figure 2. Outline of the *DeepFace* architecture. A front-end of a single convolution-pooling-convolution filtering on the rectified input, followed by three locally-connected layers and two fully-connected layers. Colors illustrate outputs for each layer. The net includes more than 120 million parameters, where more than 95% come from the local and fully connected layers.

- PKLot



# Deep Learning

- Image Translation



# Deep Learning

- Deep Fakes

## Animating Faces

A single model animates all images given only a single source image



<https://www.youtube.com/watch?v=mUfJOQKdtAk>

# Deep Learning

## Pros

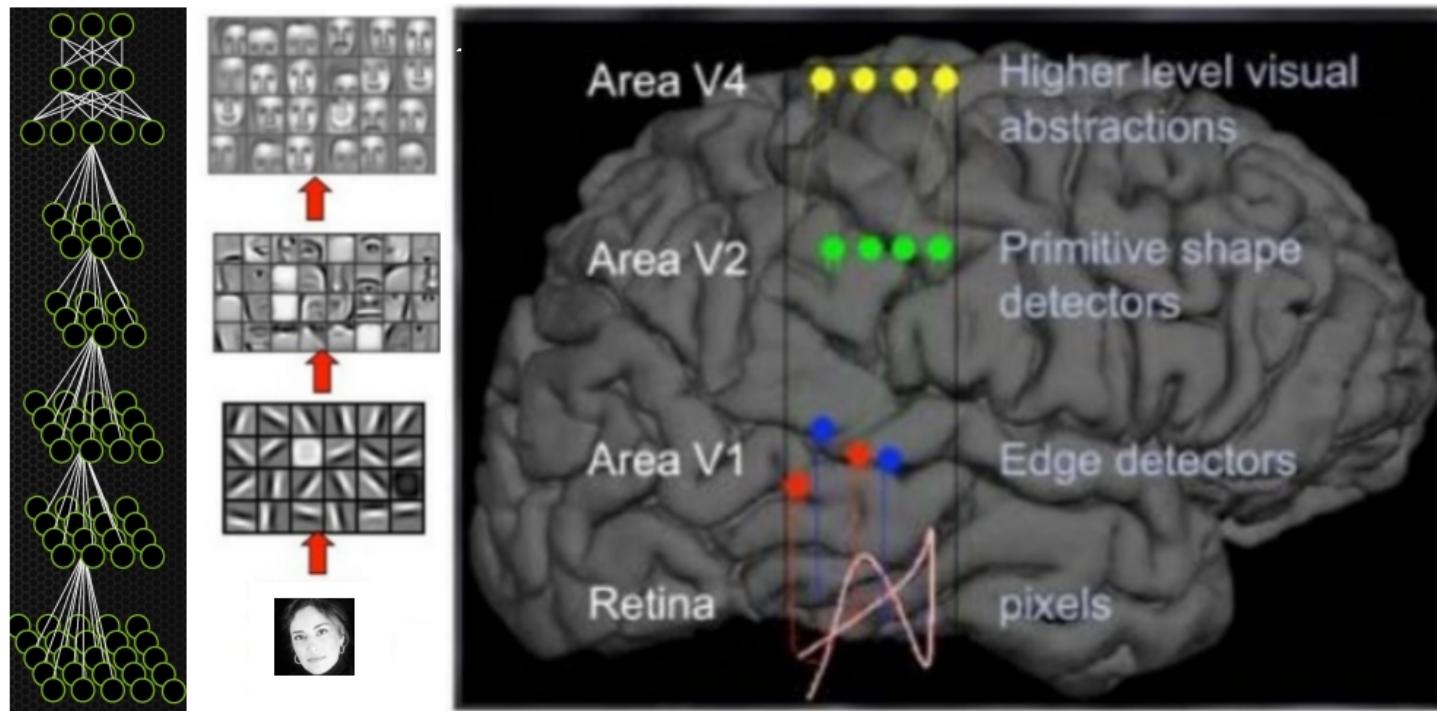
- Enables learning of features rather than hand tuning
- Impressive performance gains on
  - Computer vision
  - Speech recognition
  - Some text analysis
- Potential for much more impact

## Cons

- Computationally really expensive
- Requires a lot of data for high accuracy
- Extremely hard to tune
  - Choice of architecture
  - Parameter types
  - Hyperparameters
  - Learning algorithm
  - ...
- Computational + so many choices = incredibly hard to tune

# Convolutional Neural Networks

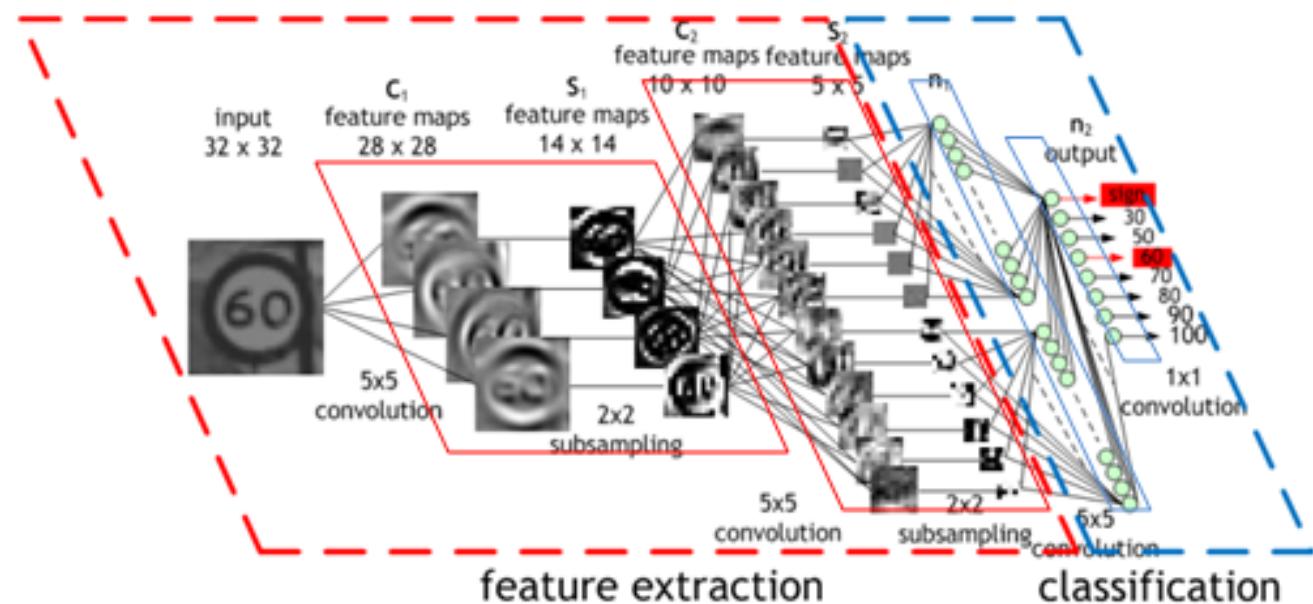
- Lecun 90's



# Convolutional Neural Networks

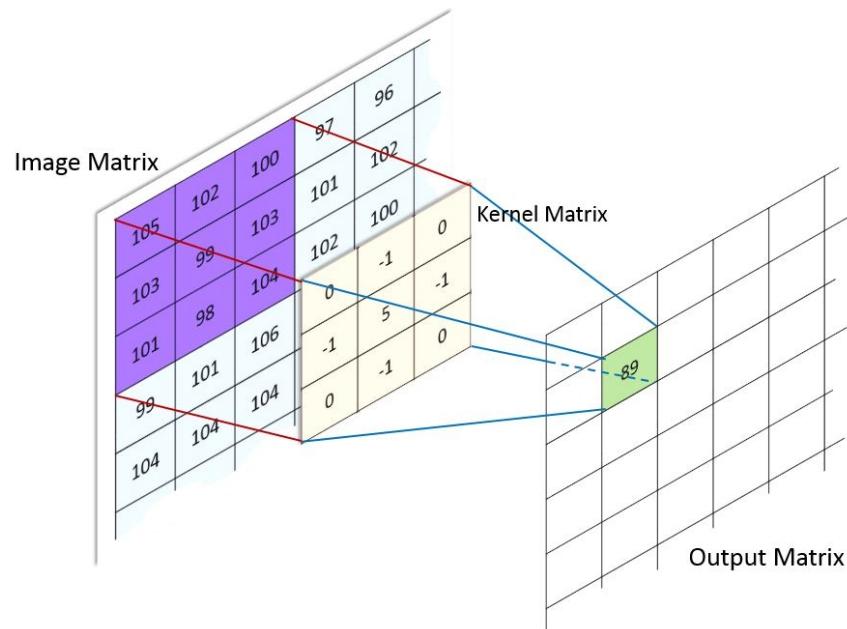
- Feature Extraction
  - Learnable Filters

- Classification
  - Fully-Connected
  - SVM
  - ....



# Convolutional Neural Networks

- Image Convolution



Input image



Convolution Kernel

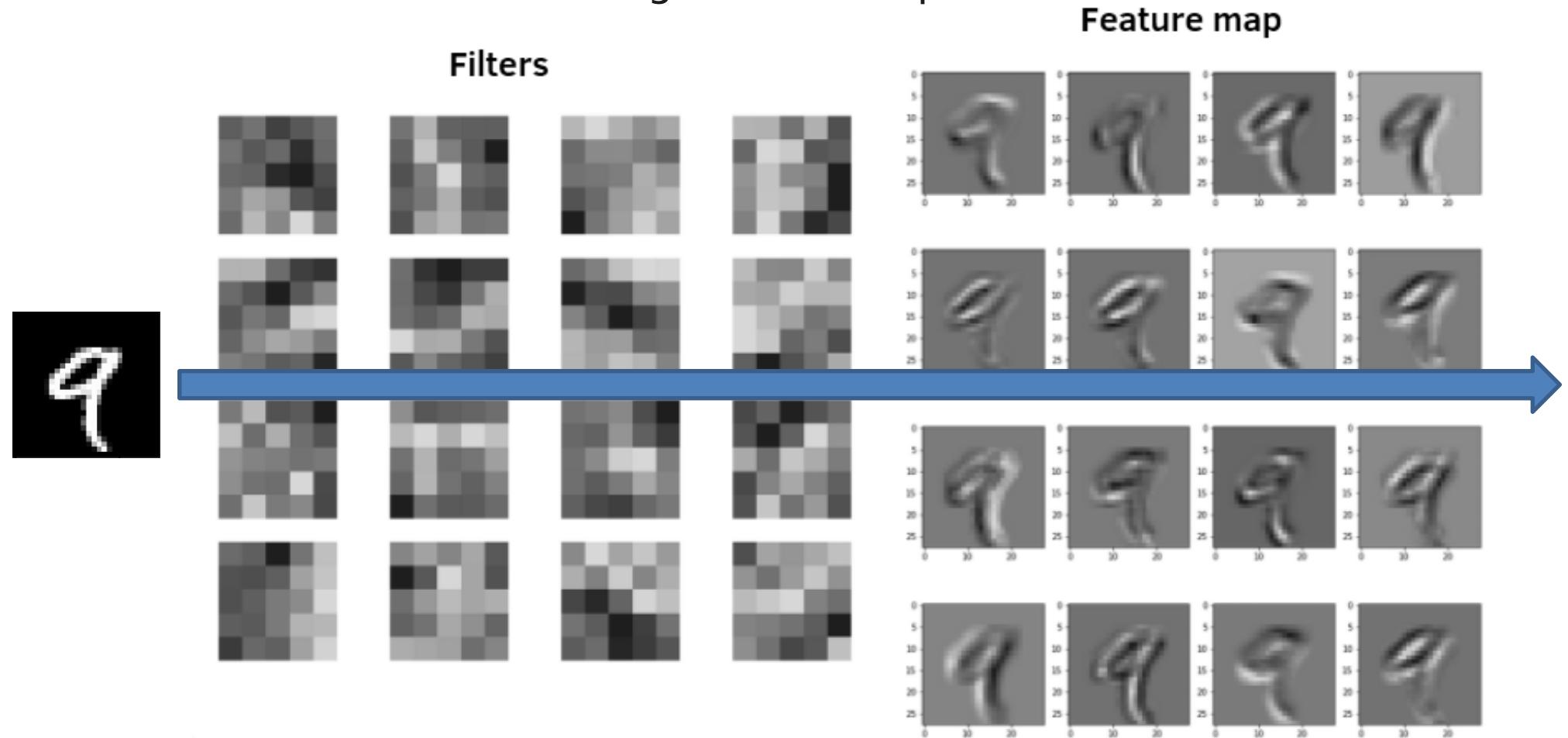
$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Feature map



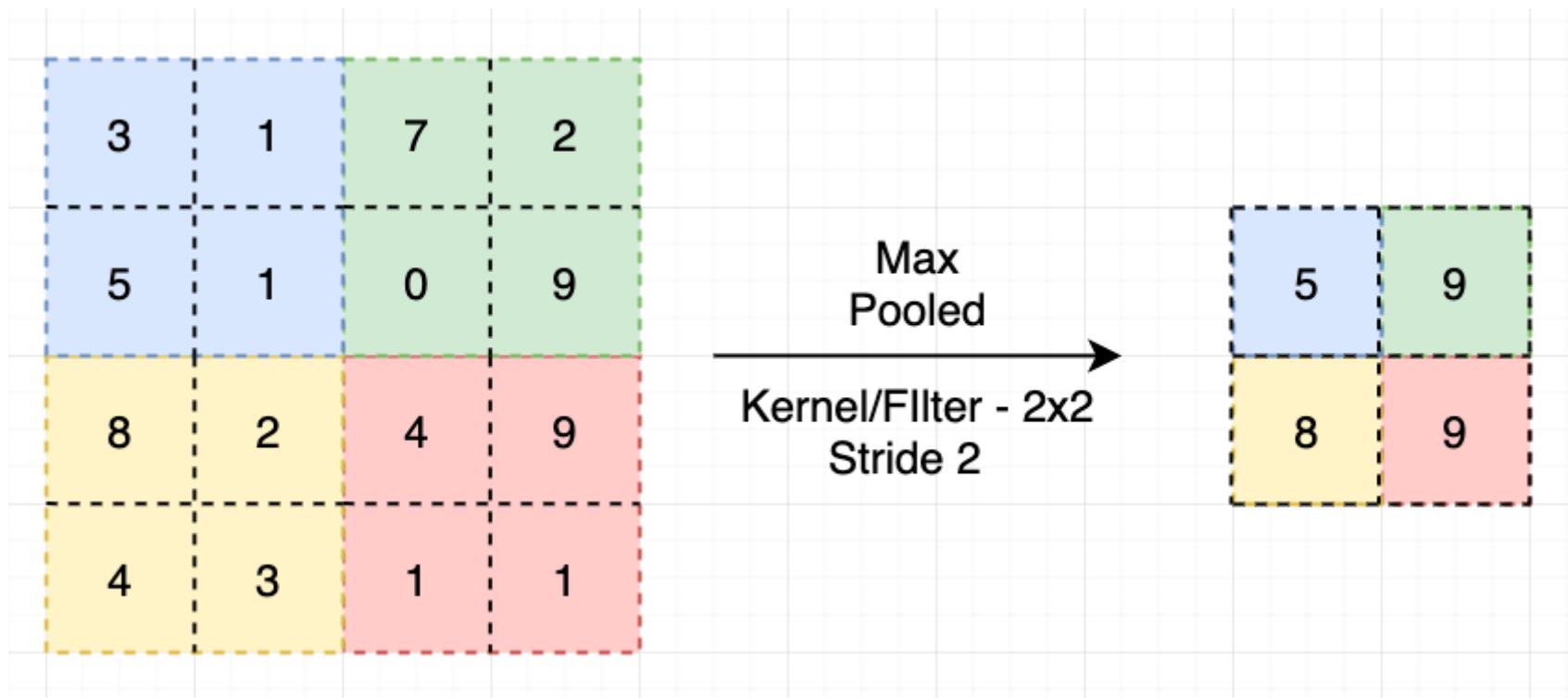
# Convolutional Neural Networks

- Learned Filters and Resulting Feature Map



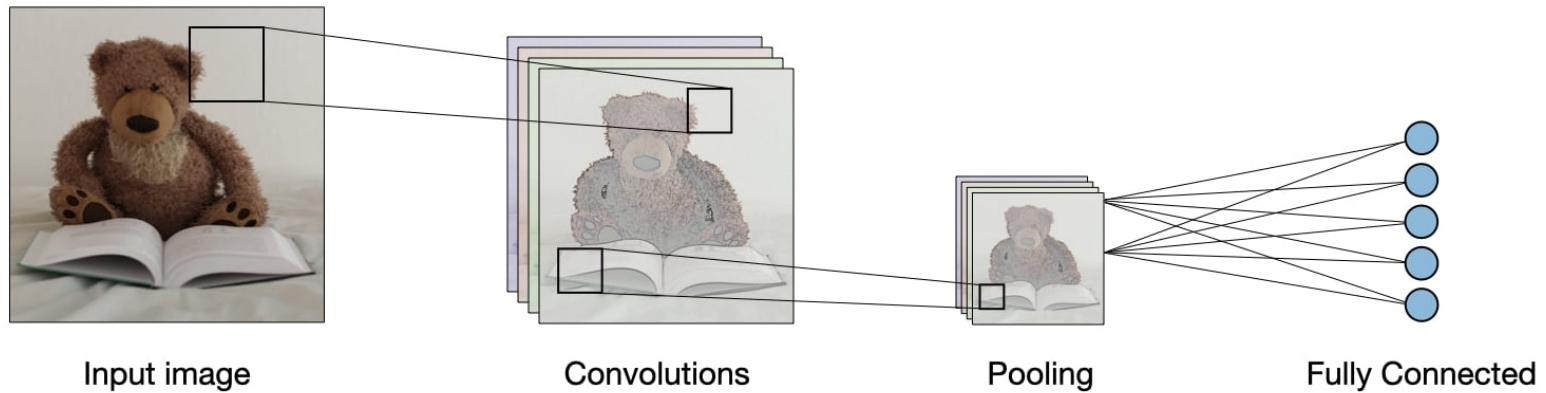
# Convolutional Neural Networks

- Pooling Layer



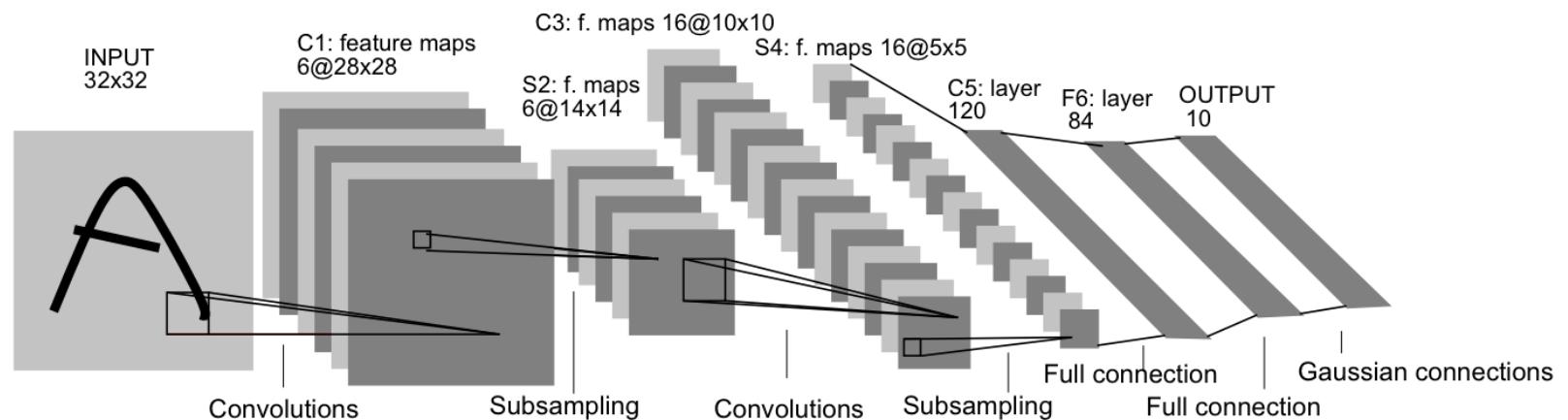
# Convolutional Neural Networks

- All Together



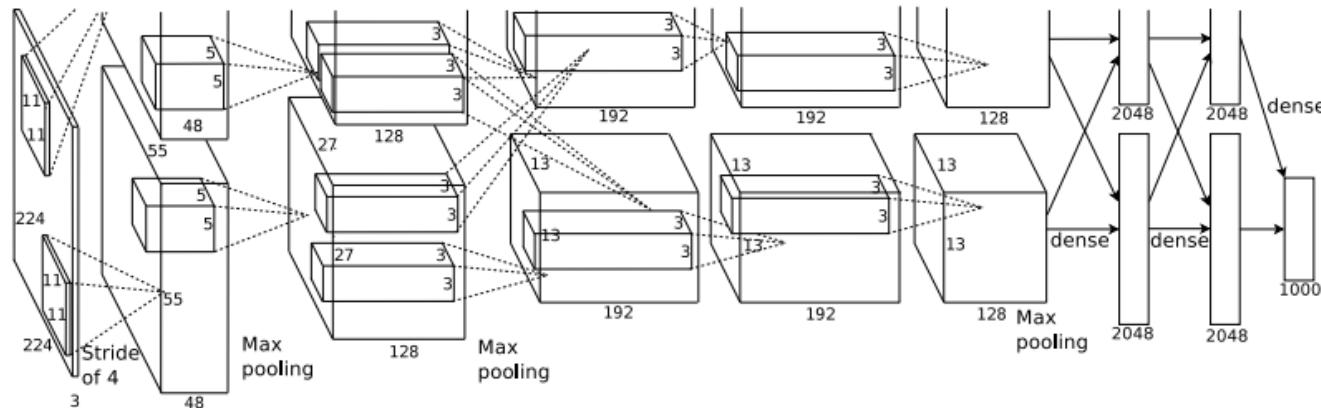
# Convolutional Neural Networks

- Lenet
  - Yan Lecun - 90's (Bell Labs / IBM / FACEBOOK)
  - Handwritten Digits
  - ~60 K Parameters
  - ~345 K Connections



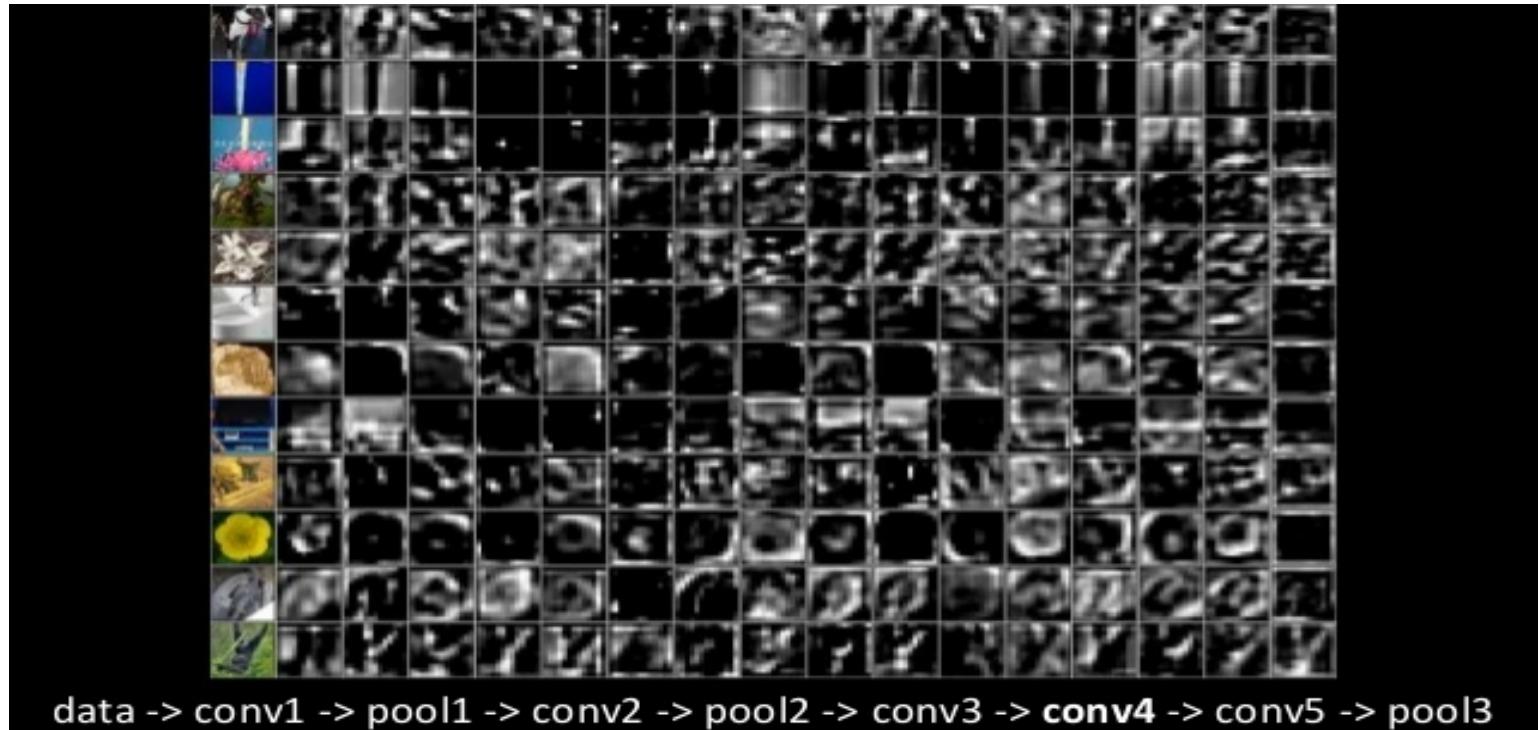
# Convolutional Neural Networks

- AlexNet
  - Alex Krizhevsky - 2012 (Krizhevsky Net)
  - Imagenet 2012 Challenge (1000 classes)  
→ 1.2 M Train, 50K Val, 150K Test
  - 2012 Winner (15.3% Error - Top 5)  
→ 2° SIFT Based (26.2%)



# Convolutional Neural Networks

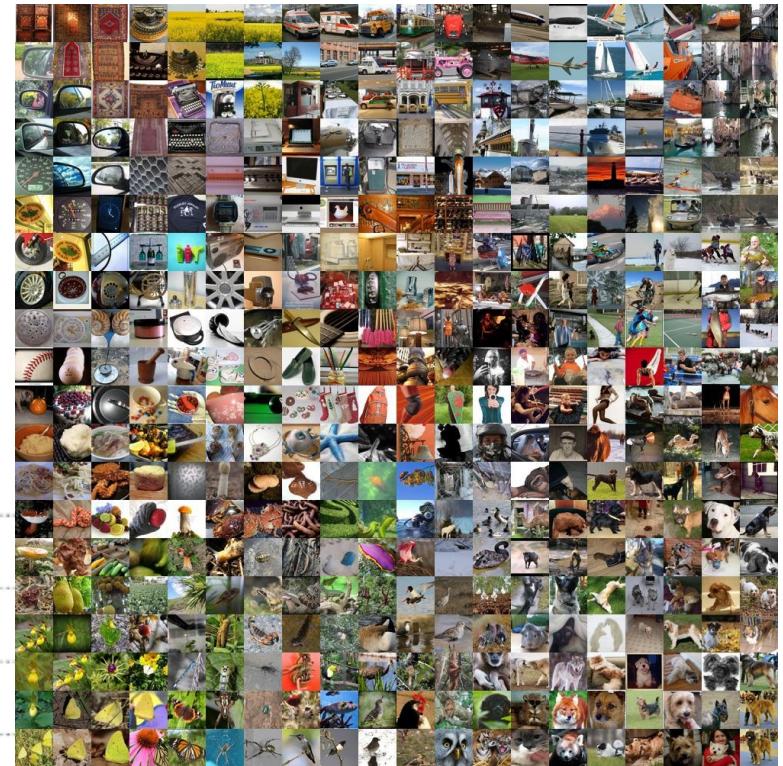
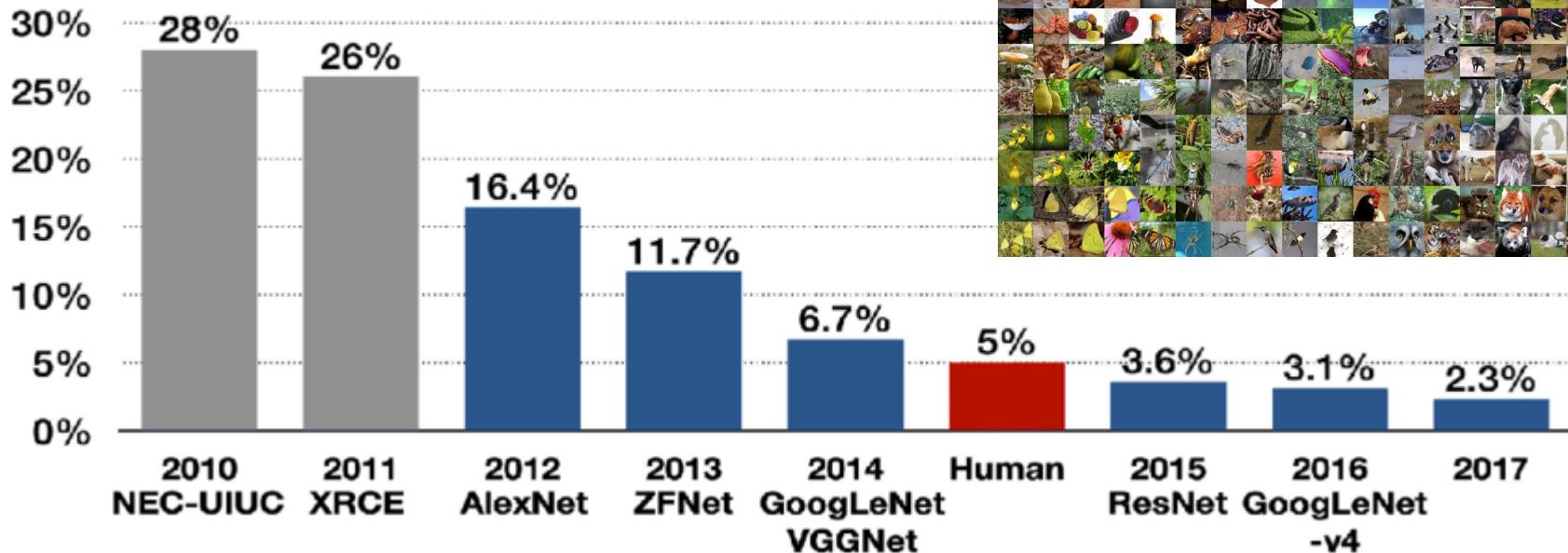
- Alexnet



# Imagenet Challenge

- Imagenet 2012 Challenge
  - 1000 classes
  - 1.2 M Train
  - 50K Val
  - 150K Test
  - [\[LINK TO ACCs\]](#)

Top-5 error



# Let's Code

- [\[LINK\]](#)